

What if?

Programmierbeleg

Beschreibung und Motivation

Der xkcd-viewer ist eine Applikation zum ansehen des xkcd-Webcomics über eine Anwendung. Sie bietet zusätzliche Funktionen wie das Favorisieren und damit Offline verfügbar machen von einzelnen Ausgaben. Zusätzlich sollen die Favoriten einen schnelleren Zugriff auf bestimmte xkcds erlauben.

Die Motivation hinter dieser Anwendung war mit einer Web-API zu arbeiten. Nach Recherche zu einfach zu verwendenden APIs, fiel mir die API von xkcd.com auf welche aus einem simplen JSON String besteht.

Probleme und Lösungsansätze

Die Aufgabe ist also nun, einen JSON String zu parsen, da aber das Hauptaugenmerk auf der bearbeitung der enthaltenen Daten liegt, wurde für diese Aufgabe ein JSON-Parser zur verwendung gezogen. Hierbei wurde der “minimal-json” Parser, welcher auf Github unter <https://github.com/ralfstx/minimal-json> zu finden ist verwendet, und als zusätzliches Package in den Sourcecode eingebunden.

Damit konnte die Arbeit an dem eigentlichen Programm beginnen, das heißt die Darstellung und Navigation durch die einzelnen Ausgaben des Webcomics. Anfänglich wurde für die Darstellung ein einfaches Image-Panel verwendet, dabei stellte sich aber schnell heraus, dass die unterschiedlich großen Formate zum Problem werden. Vorallem Ausgaben wie #1979 oder #1024 wurden bis zur unleserlichkeit skaliert. Eine Lösung wurde in Form der Scrollpane gefunden, diese wurde mit einer Stackpane gefüllt, sodass das Bild zentriert erscheint.

Das größte Problem war aber das Offlineverhalten der Anwendung, die erste Version der Anwendung stürzte direkt beim Start ab, falls keine Internetverbindung hergestellt werden konnte. Darauf hin gab es vorerst einen Button mit dem man die Bilder in einem Seperaten Ordner gespeichert werden konnten. Zusätzlich wurde der Ladealgorithmus so verändert, das zuerst geschaut wurde ob das Bild offline verfügbar ist. Zu diesem Zeitpunkt waren Favoriten und gespeicherte Bilder noch getrennt. Das heißt es gab Favoriten welche offline verfügbar waren und solche die es nicht waren. Dafür gab es non-Favoriten welche offline verfügbar waren. Der erste Lösungsansatz war die gespeicherten Bilder im Ordner einzulesen und als Favoriten zu definieren. Allerdings wurden dabei die Bilduntertitel nicht mitgeladen. Nummer und Titel konnte man als Namen des Bildes speichern (Bp: “624 – Branding.jpeg”). Jedoch nicht die Bilduntertitel, da dies meist mehrere Sätze sind.

Eine neue Lösung musste gefunden werden. Eine Klasse welche die erforderlichen Informationen speichert musste her. Die Klasse XKCD. Diese besitzt eine Variable für Nummer, Titel, Untertitel und schlussendlich den Path zum Bild.

Damit konnte favorisieren und speichern zusammen gelegt werden. Die Klasse ist Serializable, das heißt sie kann per ObjectInput/-OutputStream gelesen und gespeichert werden und somit kann einfach der Zustand der Favoritenliste beim Start und Ende des Programms gespeichert/gelesen werden.

Entwurf

Der Entwurf der einzelnen Klassen besteht hauptsächlich aus einer Verteilung der Aufgaben, mit Control als zentraler Verwaltungspunkt. Damit ist es auch die Klasse welche die Liste mit den Favoriten enthält. Zusätzlich dient sie zur Kommunikation zwischen den beiden Fenstern, enthält Referenzen zu allen anderen Objekten, und gibt zum Beispiel den Befehl zum öffnen und schließen aus dem Hauptfenster an das Favoritenfenster weiter (s. `showFavorites()`). Zusätzlich gibt es die Interface-Klassen für die Beiden Fenster, `MainWindowController` und `FavoriteWindowController`, welche die Methoden der Buttons beinhalten. Der `MainWindowController` besitzt dazu noch den Algorithmus welcher das Bild, sowie die Textdaten lädt. Dieser erstellt sich hierfür eine Instanz der Parser-Klasse, welche den API String durch eine statische Methode der `URLReader`-Klasse lädt. Sollte keine Internetverbindung bestehen, oder `isInternet` auf `false` gesetzt sein. Werden stattdessen die `OfflineLoader`-Methoden aufgerufen welche die zu ladende Nummer herausfinden und an `loadOffline()` zurückgeben. Diese holt sich die Textdaten sowie den Bild-Path von der Speicherklasse.

Alternativ kann der Ladeprozess auch durch einen Klick auf den Eintrag im Favoriten-Fenster gestartet werden.

Fazit

Der jetzige Stand der Programms ist funktionsfähig und hat die wichtigsten Features. Es ist gegen einfache Fehler, wie zum Beispiel `ArrayIndexOutOfBoundsException` oder `FileNotFoundException` gesichert und zeigt in diesen Fällen eine Fehlermeldung.

Jedoch ist noch viel unnötiger Code, aus Beispielsweise alter Fehlerbehandlung vorhanden. Dieser wird nicht mehr ausgeführt, macht den Code aber noch unleserlicher als er sowieso schon ist.

Der Code ist ziemlich unstrukturiert da beim Start noch kein klarer Plan stand und Features später hinzugefügt wurden, welche umstrukturierungen benötigten.

Geplante Features

Einige Features waren noch geplant, konnten aber nicht mehr umgesetzt werden, so zum Beispiel ein Button welcher eine Erklärung zum aktuell gezeigten xkcd aufruft. Jedoch hängt sich das Programm auf, sobald er betätigt wird und wurde somit deaktiviert.

Außerdem sollte zukünftig ein Wechsel vom Offlinemodus in den Onlinemodus möglich sein, dafür müsste man die Modi klarer voneinander abtrennen (nicht nur eine `bool`-Variable, sondern unterschiedliche Funktionen).

Letztlich sollte die UI an die Vorlieben des Nutzers anpassbar sein. Das heißt man sollte die Farben anpassen können, oder Tastenkürzel mit Funktionen belegen.

Anleitung

Beim Start der Anwendung wird der neuste xkcd geladen, sollte keine Internetverbindung bestehen wird stattdessen der zuletzt favorisierte xkcd geladen.

Danach kann entweder über den Navigationsblock unten rechts, oder für einfachere Benutzung der Nummernblock zur Navigation verwendet werden. Hierbei lädt NUM_4 den vorherigen NUM_6 den nächsten, NUM_5 einen zufälligen und NUM_8 den neusten xkcd. Falls man eine bestimmte Nummer anwählen möchte kann man den Fokus mit NUM_0 auf das Textfeld legen und mit Enter bestätigen. Die vorlaufende Null wird beim Nummerparsen ignoriert.

Während der Fokus auf der Scrollpane liegt kann man mit den Pfeiltasten scrollen und mit Enter einen Favoriten hinzufügen/entfernen.

Ein Klick auf den “Favoriten ein-/ausklappen” öffnet bzw. schließt das Favoritenfenster.

Im Favoritenfenster kann auf einen Eintrag in der Tabelle geklickt werden um diesen xkcd zu laden.

Die Favoritenliste wird beim schließen automatisch gespeichert, kann aber auch manuell durch den “save” Button gesichert werden. Der “load” Button lädt die Liste erneut aus dem Savefile, damit können Änderungen der aktuellen Sitzung rückgängig gemacht werden.

Schlussendlich ist es möglich das Programm in den Offlinemodus zu schalten, um die Navigation auf die Favoriten zu beschränken. Wenn bei Start der Anwendung keine Verbindung zum Internet hergestellt werden konnte, ist diese CheckBox ausgegraut und der Haken ist drin.

Verwendete Hilfsmittel:

- JSON-Parser: <https://github.com/ralfstx/minimal-json>
- Stackoverflow (a lot)

"Ich erkläre, dass ich die vorliegende Arbeit selbständig und nur unter Verwendung der angegebenen Literatur und Hilfsmittel angefertigt habe."